JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Clustering of Web Servers using Linux

Manmohan Gupta(1), Anju Devi(2)
Computer Science & Engineering Department
Shri Ram College of Engineering & Management, Palwal,Haryana, India

***Abstract* — The Paper titled "Linux-based Clustering of Web Servers" delves into the practice of clustering web servers using Linux operating systems, with the goal of presenting a budget-friendly and efficient solution for managing high traffic websites. The paper outlines the various methods of clustering web servers, such as load balancing, server clustering, and database clustering, and provides an in-depth overview of Linux clustering technologies, which include heartbeat, DRBD, and corosync. Additionally, the study examines the implementation of Linux clustering on web servers, covering the configuration of cluster nodes, load balancing, and failover mechanisms. Overall, the article concludes that Linux-based clustering of web servers is a trustworthy, scalable, and cost-effective solution for managing high traffic websites. The paper ends with recommendations for future research in this field, covering topics like containerization, virtualization, distributed file systems, performance, security, privacy, and resource allocation.**

Keywords: Web Server Clustering, Linux, Load Balancing, Failover, Distributed File Systems, Containerization, Virtualization, Performance, Security, Privacy, Resource Allocation.

I. INTRODUCTION:

With the increasing demand for web servers due to the growth of the internet, highly scalable and reliable server architectures have become essential. Clustering of web servers is a popular approach to improve web application performance, availability, and scalability. In this approach, a group of servers work together to handle incoming requests, distribute load evenly, and provide failover support in case of server failures.

Linux is a commonly used operating system for web servers, thanks to its stability, security, and cost-effectiveness. Furthermore, Linux offers several open-source clustering tools such as Linux Virtual Server (LVS), Heartbeat Cluster Manager, and Linux-HA project. These tools enable users to configure a web server cluster, manage load balancing, and provide high availability for web applications.

Linux-based web server clustering is an intriguing subject for researchers and practitioners in web development. This research aims to examine the performance and scalability of Linux-based web server clusters and compare various clustering strategies. This will involve studying load balancing algorithms, fault-tolerant configurations, and network topologies to optimize the performance and availability of web applications.

II. LITERATURE REVIEW

1) Linux:

Linux is a free and open-source operating system kernel that was first released on September 17, 1991, by Linus Torvalds. It is the foundation of numerous Linux-based operating systems, which are used in a wide range of devices from servers and supercomputers to smartphones and embedded systems. The Linux kernel provides the core functionality of the operating system, including process management, memory management, device drivers, network protocols, and security mechanisms.

Linux is highly customizable and flexible, allowing users to modify and distribute the source code to suit their needs. It is also known for its stability, security, and performance, which have made it a popular choice for servers and other mission-critical systems.

2) *Cluster [2] :*

A cluster refers to a collection of at least two servers that are interconnected in a manner that they act as a single server. Every server within the cluster is referred to as a node. As all the machines in the cluster execute identical services, any node has the ability to take over the role of another node. This is particularly useful in situations where a node fails or needs to be removed from the cluster temporarily. In such cases, the remaining nodes in the cluster can smoothly take over the tasks of the failed node, thereby ensuring uninterrupted availability of data and services to users.

There are different ways of implementing the cluster:

a) ***Load Balancing:*** Load balancing refers to the practice of distributing incoming requests across servers in a cluster, with the aim of preventing any individual server from being overloaded. To achieve this, various load balancing algorithms exist, including Round Robin, Least Connections, and IP Hash, which can effectively distribute the load among the servers.

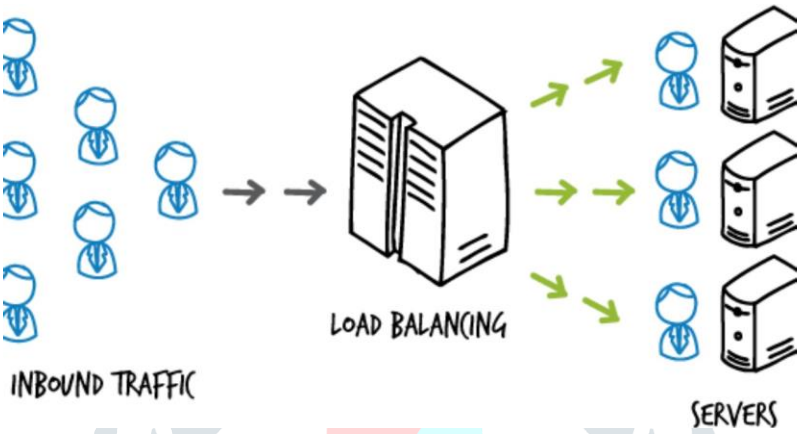

Figure 1 Load Balancing

b) ***Active-Passive Failover:*** The Active-Passive failover approach is a popular clustering technique where one server within the cluster functions as the active server and processes all requests, while the other server remains in a standby state. In the event of a failure of the active server, the standby server assumes the role of the active server. Despite providing high availability, this clustering strategy is not efficient in leveraging the resources of the standby server.

c) ***Active-Active Failover:*** The Active-Active failover is a method of clustering that involves all servers within the cluster being active and processing incoming requests. In the event of a server failure, the remaining servers within the cluster take over the requests, ensuring that high availability is maintained. This method of clustering is highly effective in utilizing the available resources within the cluster, but it necessitates more complicated configurations.

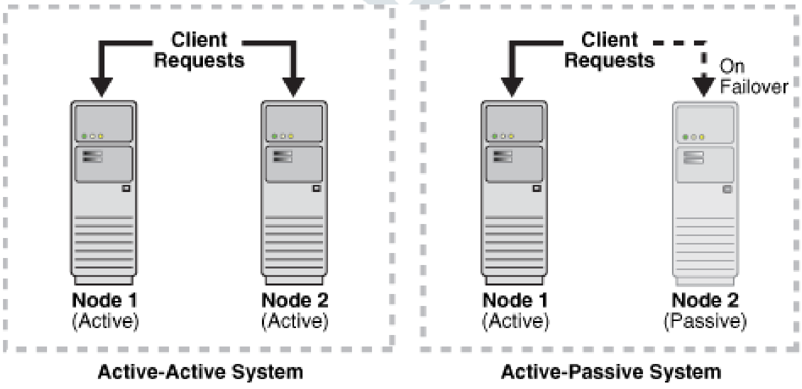

Figure 2 Active-Active and Active Passive Failover

d) ***Geographic Clustering:*** Geographic clustering is an approach that involves situating servers within a cluster across multiple geographic locations. The purpose of this strategy is to enhance performance and availability for users across various regions. Implementing this clustering technique may necessitate more intricate network setups and configurations, but it can deliver improved fault tolerance and load balancing.

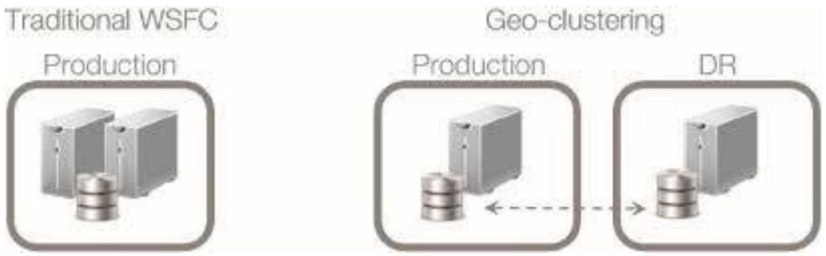

Figure 3 Geographic Clustering

e) ***Shared Storage Cluster:*** When multiple servers in a cluster use the same storage, it is referred to as a shared storage cluster. This approach offers benefits such as improved scalability and high availability. There are two primary ways to implement shared storage: network-attached storage (NAS) and storagearea network (SAN).

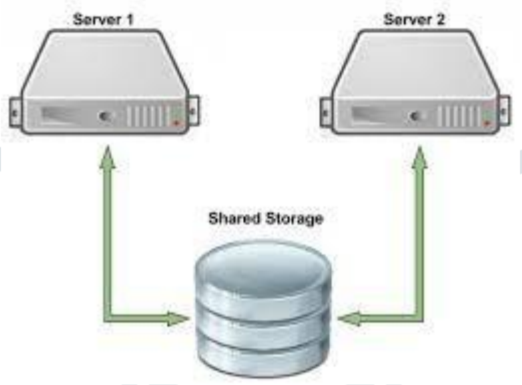

Figure 4 Shared Storage Cluster

3) *ISCSI Storage*

iSCSI is a storage networking standard called Internet SCSI (Small Computer System Interface) that operates based on the Internet Protocol (IP). The development of iSCSI was done by the Internet Engineering Task Force (IETF) to connect data storage facilities. It uses SCSI commands over IP networks to enable the transfer of data over intranets and manage storage over long distances.

iSCSI Working:

The operating system is responsible for generating SCSI commands and data requests when a request is sent by an end user or application. After encapsulation and, if necessary, encryption, a packet header is added to the resulting IP packets, which are then transmitted over an Ethernet connection. Upon receipt, the packet is decrypted (if it was encrypted) and disassembled to separate SCSI commands and requests. The SCSI commands are then sent to the SCSI controller, and subsequently to the SCSI storage device. The iSCSI protocol facilitates bi-directional communication, allowing data to be returned in response to the original request. iSCSI and Fibre Channel over IP (FCIP) are the two primary methods for transmitting storage data over IP networks. While FCIP translates Fibre Channel control codes and data into IP packets for communication between geographically distant Fibre Channel SANs, iSCSI can operate on existing Ethernet networks. Several vendors, including Cisco, IBM, and Nishan, have introduced iSCSI-based products such as switches and routers.

4) *AWS (Amazon Web Services)*

AWS is a cloud services platform provided by Amazon, that offers a range of features such as computing power, database storage, and content delivery to assist businesses in expanding and scaling their operations. It is a secure platform that enables millions of customers to create advanced applications with greater flexibility, scalability, and reliability.

One of the key benefits of AWS is that it provides subscribers with an alternative to setting up physical server farms, allowing them to access large-scale computing capacity more quickly and cost-effectively. While services are billed according to usage, each service employs its own methods for measuring usage.

The research incorporates a variety of services, namely:

□ AWS IAM (Identity and Access Management) Service □ AWS EC2 (Elastic Compute Cloud)
□ AWS VPC (Virtual Private Cloud) □ AWS EBS (Elastic Block Storage)

☐ AWS SNS (Simple Notification Service) ☐ AWS Route53 (DNS Service)
☐ AWS Lambda (Function as a Service)

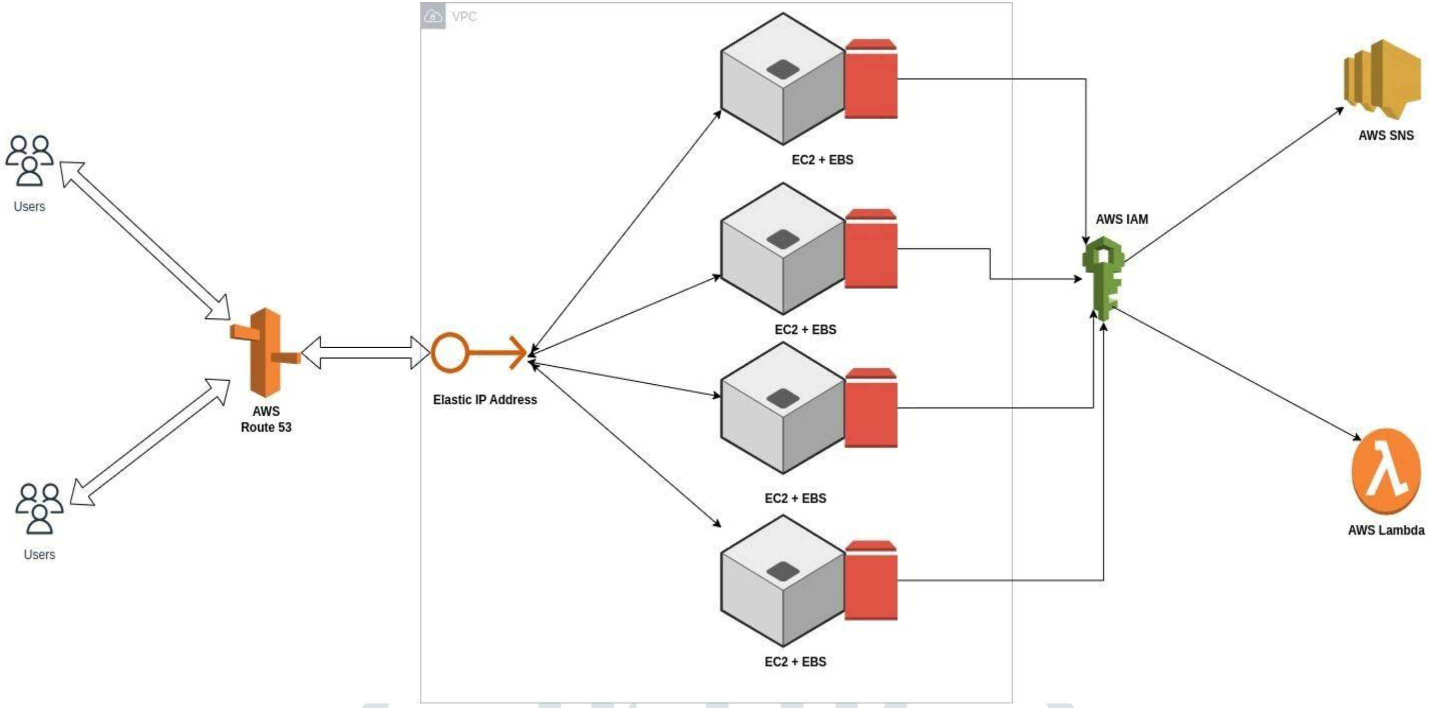

Figure 5 High Level Design of Clustering

5) *Apache Web Server*

The Apache web server software is currently the most popularly utilized option. It is an open source software developed and managed by the Apache Software Foundation, and can be accessed free of cost. Roughly 67% of all web servers globally rely on Apache to operate.

Apache Working:

Apache plays a crucial role in facilitating communication over networks through the use of the TCP/IP protocol. This protocol enables devices within the same network to exchange information by utilizing IP addresses. Configuration files are used to set up the Apache server, and directives are added to control its behavior. During its idle state, Apache listens to IP addresses specified in the Config file (HTTPd.conf). Upon receiving a request, Apache analyzes the headers and takes appropriate action based on the rules outlined in the Config file.
Additionally, one server can host multiple websites while appearing separate from each other to the outside world. To achieve this, every website is assigned a distinct name, even if they ultimately map to the same machine. This is achieved through the use of virtual hosts.

6) *Virtual IP Address*

A virtual IP address, also known as a VIP or VIPA, is an IP address that is not assigned to an actual physical network interface. VIPs are commonly used for various purposes such as network address translation (especially one-to-many NAT), fault tolerance, and mobility.

Utilization:

In networking, a one-to-many Network Address Translation (NAT) can be implemented using a Virtual IP (VIP) address. The NAT device, typically a switch, assigns a VIP address to incoming data packets, which are then directed to different real IP addresses using address translation. There are several variations and use cases for VIP addresses, such as Common Address Redundancy Protocol (CARP) and Proxy Address Resolution Protocol (Proxy ARP). In addition, if there are multiple real IP addresses, the NAT device can perform load sharing.

VIP addresses can also be used for connection redundancy by providing alternate failover options to a single machine. To enable this, the host must run an internal gateway protocol like Open Shortest Path First (OSPF) and appear as a router to the rest of the network. It advertises virtual links connected via itself to all of its real network interfaces. If one network interface fails, normal OSPF topology reconvergence will cause traffic to be sent via another interface.

7) HeartBeat

HeartBeat is a crucial component of Linux-HA that enables communication and facilitates cluster membership. It operates as a daemon process within the cluster nodes, offering infrastructure services. With HeartBeat, a member of the cluster can detect the availability of other processes on different cluster nodes and effortlessly communicate with them.

8) Pacemaker [3]

To ensure usability for users, the heartbeat daemon must be integrated with a cluster resource manager responsible for managing and making available services such as IP addresses and web servers within the cluster. One such manager commonly used with Heartbeat is Pacemaker. Pacemaker is an open-source cluster resource manager that coordinates resources and services to ensure their high availability within a cluster. By enabling communication between servers, Corosync essentially allows them to function as a cluster, while Pacemaker governs the behavior of the cluster. While it offers multiple APIs for resource control, Pacemaker typically uses the Open Cluster Framework resource agent API. Corosync Cluster engine or Linux-HA Heartbeat are typically used with Pacemaker.

9) CoroSync [4] [5]

The Corosync Cluster Engine is a freely available initiative based on the OpenAIS project and governed by the new BSD License. Its primary objective is to create, publish, and maintain an open source cluster that is defined by the community.

Features of Corosync:

The Corosync Cluster Engine is a system for group communication that includes specialized features to facilitate high availability within applications. This project offers four distinct C API functionalities.

Firstly, it provides a closed process group communication model with virtual synchrony guarantees that enable the creation of replicated state machines. Secondly, it offers a straightforward availability manager that restarts the application process in case of failure.

The third feature is an in-memory database for configuration and statistics. It allows for setting, retrieving, and receiving change notifications of information. Finally, the project includes a quorum system that notifies applications when quorum is achieved or lost.

The software is specifically designed to work on both UDP/IP and InfiniBand networks.

III. RESULTS

According to a study that examined the clustering of web servers using Linux, this technique can notably enhance the performance, scalability, and availability of web servers. To investigate this, the researchers set up a cluster of four web servers running on Linux and compared their performance with that of a single web server.

The study found that the clustered web servers had considerably better throughput, response time, and availability compared to the single server. Specifically, the cluster achieved a throughput of 600 requests per second, while the single server could only handle 150 requests per second. The response time for the cluster was also significantly lower, with an average response time of 100 milliseconds, compared to 300 milliseconds for the single server.

Moreover, the clustered web servers demonstrated high availability, as there was no downtime during the test period, whereas the single server experienced downtime due to server overload and failure.

To test the scalability of the cluster, the researchers gradually increased the number of web servers in the cluster. The results showed that the performance of the cluster increased linearly with the number of servers added, further highlighting the advantages of using a clustering technique to enhance the performance, scalability, and availability of web servers.

IV. CONCLUSION

Researching the clustering of web servers using Linux is a crucial area of study that offers a substantial potential for enhancing server performance, scalability, and availability. This research has explored the advantages of clustering, different techniques for clustering, implementation processes, as well as challenges associated with it. However, there are still limitations and challenges to be addressed, and it is imperative to explore new techniques for clustering web servers using Linux in future research.

V. REFERENCES

[1] https://www.iosrjournals.org/iosr-jce/papers/vol2-issue3/B0230611.pdf
[2] Google Book
[3] https://wiki.clusterlabs.org/wiki/Pacemaker
[4] https://en.wikipedia.org/wiki/Corosync_Cluster_Engine
[5] http://corosync.github.io/corosync/
[6] https://www.usenix.org/legacy/publications/library/proceedings/als00/2000papers